Score-, Wald- and likelihood ratio tests

gean piere ventura cruz

2022-10-01

Task 1:

we will be using the code from the "Likelihood, numerical optimization and the Bootstrap" project in order for us to estimate the z-values with help of the wald statistic and then compare these with the glm table.

```
summary(modell)
```

```
##
## Call:
## glm(formula = Resultat ~ Alder + Kon + Utbildare, family = "binomial",
##     data = data_individ)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5786  -0.9712  -0.7555   1.2398   2.0089
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)           0.336619   0.254014   1.325    0.185
## Alder                -0.045140   0.009185  -4.914 8.91e-07 ***
## KonMan                0.330364   0.135873   2.431    0.015 *
## UtbildareTrafikskola  1.052320   0.157491   6.682 2.36e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1355.9  on 999  degrees of freedom
## Residual deviance: 1271.5  on 996  degrees of freedom
## AIC: 1279.5
##
## Number of Fisher Scoring iterations: 4
```

```
wald_statistics <- theta_est/standard_error
wald_statistics
```

```
## [1]  1.325185 -4.914361  2.431413  6.681763
```

This indeed matches our z-values of the output,

Task 2:

in this part we will use the generalized likelihood ratio statistics that corresponds to the wald statistics in the first task and then determine the p values.

```r
#the order is interceopt, alder, kon, utbvildning

eta_intercept <- NR(theta0 = c(0, 0, 0), niter = 10, y = y, X = X[, -1])
theta_intercept <- c(0,eta_intercept)
#theta_intercept

eta_alder <- NR(theta0 = c(0, 0, 0), niter = 10, y = y, X = X[, -2])
theta_alder <- c(eta_alder[1],0,eta_alder[2],eta_alder[3])
#theta_alder

eta_kon <- NR(theta0 = c(0, 0, 0), niter = 10, y = y, X = X[, -3])
theta_kon <- c(eta_kon[1],eta_kon[2],0,eta_kon[3])
#theta_kon

eta_utbildare <- NR(theta0 = c(0, 0, 0), niter = 10, y = y, X = X[, -4])
theta_utbildare <- c(eta_utbildare,0)
#theta_utbildare


L_est_theta_intercept <- 2*log(L(theta_est,y,X,1000)/L(theta_intercept,y,X,1000))
#L_est_theta_intercept
L_est_theta_alder <- 2*log(L(theta_est,y,X,1000)/L(theta_alder,y,X,1000))
#L_est_theta_alder
L_est_theta_kon <- 2*log(L(theta_est,y,X,1000)/L(theta_kon,y,X,1000))
#L_est_theta_kon
L_est_theta_utbildare <- 2*log(L(theta_est,y,X,1000)/L(theta_utbildare,y,X,1000))
#L_est_theta_utbildare

pchisq(L_est_theta_intercept,df = 1, lower.tail = FALSE)
```

```
## [1] 0.1840288
```

```r
pchisq(L_est_theta_alder,df = 1, lower.tail = FALSE)
```

```
## [1] 3.837442e-07
```

```r
pchisq(L_est_theta_kon,df = 1, lower.tail = FALSE)
```

```
## [1] 0.01467596
```

```r
pchisq(L_est_theta_utbildare,df = 1, lower.tail = FALSE)
```

```
## [1] 1.141609e-11
```

The corresponding p-values seem to be quite similar to our output. The p values for theta_alder and theta_utbildare are a slightly bit different to the ones from the output. We denote that the wald statistic is asymptotically standard normal distributed and the likelihood ratio is chi2 distributed with 1 degree pf freedom. If you would take a standard normalised variabel X and square it, then it turns out to be chi2 distributed.

Task 3:

In this part we are going to Compute the ML estimate of the variables $(\theta_{Alder}, \theta_{Utbildare})$ under $H0: \theta = (\theta_{intercept}, \theta_{Kon}) = (0, 0)$ and use this to determine a P-value based on the generalized score statistic.

```
eta_test <- NR(theta0 = c(0, 0), niter = 10, y = y, X = X[, -c(1,3)])
#eta_test
theta_alder_utbildare <- c(0, eta_test[1], 0, eta_test[2])
theta_alder_utbildare
```

```
## [1]  0.00000000 -0.02627326  0.00000000  1.10314627
```

```
score_statistic <- t(S(theta_alder_utbildare,y,X,1000))%*%solve(I(theta_alder_utbildare,y,X,1
000))%*%S(theta_alder_utbildare,y,X,1000)
score_statistic
```

```
##          [,1]
## [1,] 10.84141
```

```
pchisq(score_statistic, df = 2, lower.tail = FALSE)
```

```
##             [,1]
## [1,] 0.004424022
```

We expand our eta vector using the null hypothesis to obtain theta_alder_utbildare. This vector we then use to compute the generalized score statistic, 10.84…, which then we use to compute a p value for it and gives us 0.0044.

Task 4:

In this last part we are going to compute the profile likelihood on a suitable grid of parameter values then use these to graph the profile likeli hood together with the corresponding estimated likelihood. We will also decidea 95% confidence interval based on the profile likelihood visually from the figure by drawing a horizontal line at a suitable level

```r
theta.Kon <- seq(-1,1,0.01) # example value
#theta.Kon

profil_theta_likelihood <- c()
for (i in 1:length(theta.Kon)) {
  new_theta <- c(0,0,theta.Kon[i],0)
  profil <- glm.fit(x = X[, -3], y = y,
                    offset = theta.Kon[i] * X[,3],
                    family = binomial())
  new_theta[c(1,2,4)] <- profil$coeff
  profil_theta_likelihood[i] <- L(new_theta,y,X,1000)/L(theta_est,y,X,1000)
}
#profil_theta_likelihood


estimated_theta_likelihood <- c()
for (i in 1:length(theta.Kon)) {
  new_nr_theta <- c(theta_est[1],theta_est[2],0,theta_est[4])
  new_nr_theta[3] <- theta.Kon[i]
  estimated_theta_likelihood[i] <- L(new_nr_theta,y,X,1000)/L(theta_est,y,X,1000)
}



#estimated_theta_likelihood

plot(theta.Kon, profil_theta_likelihood,type = "l",col= "red")

points(theta.Kon, estimated_theta_likelihood, type = "l", col = "blue" )

abline(h = exp(-1/2 * qchisq(.95, df=1)), lwd = 2, col = "green")
```
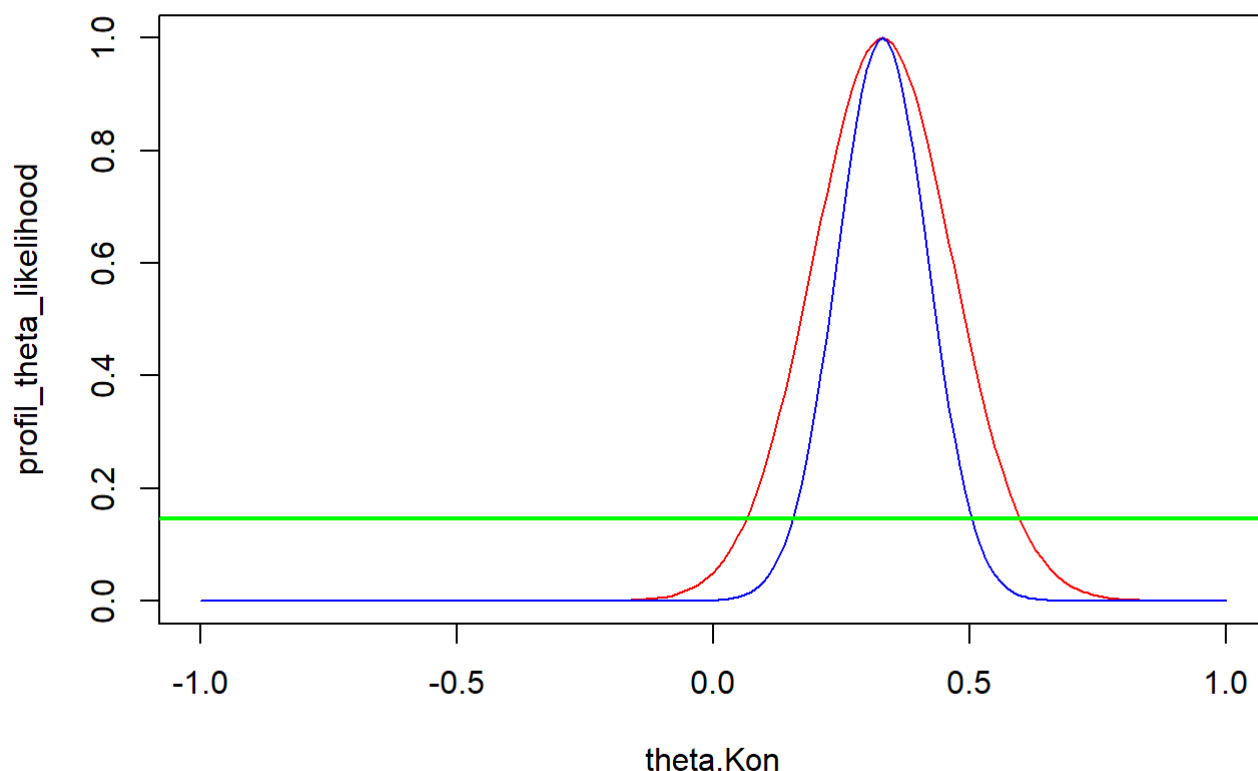
```
#exp(-1/2 * qchisq(.95, df=1))
#1.05*0.135873+0.330364 upper limit for wald
#-0.39*0.135873+0.330364 lower limit for wald
```

The blue line is the visualization of the estimated likelihood and the red represents the profile likelihood. The green horizontal line represents the 95% confidence interval based on the profile likelihood and it is calculated by using the expression that is marked as a comment in the code chunk above (this is also a standard way of calculating the 95% confidence interval). We can observe visually that theta_kon is approximate 0.4 and lies in the interval of (0.1, 0.6). We then calculate a confidence interval for the wald test using the standard error of 0.135873 and the theta_kon of 0.330364, and we obtain the interval (0.27,0.47) approximated which is a accurate interval to the profile likelihood.